IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of:	
Applicant(s): Edward D. Glas *et al.*	Examiner: Phuong Huynh
Serial No: 10/810,944	Art Unit: 2857
Filing Date: March 26, 2004	

Title: LOAD TEST SIMULATOR

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Dear Sir:

    Applicants submit this brief in connection with an appeal of the above-identified patent application. Payment of the $10.00 difference between the current fee for filing this Appeal Brief and a fee paid in connection with an appeal filed May 15, 2007, is submitted herewith. Since the aforementioned Appeal Brief never reached the Board, applicants need not pay a new fee for the subject Appeal Brief. In the event any additional fees may be due and/or are not covered by the fee submission, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1063 [MSFTP637US].

I.   Real Party in Interest (37 C.F.R. §41.37(c)(1)(i))

The real party in interest in the present appeal is Microsoft Corporation, the assignee of the present application.

II.   Related Appeals and Interferences (37 C.F.R. §41.37(c)(1)(ii))

Appellants, appellants' legal representative, and/or the assignee of the present application are not aware of any appeals or interferences which may be related to, will directly affect, or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III.   Status of Claims (37 C.F.R. §41.37(c)(1)(iii))

Claims 1-21 stand rejected by the Examiner. The rejection of claims 1-21 is being appealed.

IV.   Status of Amendments (37 C.F.R. §41.37(c)(1)(iv))

No amendments were submitted after the Final Office Action. (*See* Applicants' Reply to Final Office Action dated February 25, 2008).

V.   Summary of Claimed Subject Matter (37 C.F.R. §41.37(c)(1)(v))

A.   Independent Claim 1

Independent claim 1 relates to a computer-implemented system that test loads a server comprising a dynamic load adjustor component that dynamically adjusts user characteristics based at least in part on a browser type, for distribution thereof as a percentage of total requests sent to a server being load tested. By dynamically adjusting the user characteristics, the system can simulate different users' behavior, and therefore different operating environments for the load test. (*See* Fig. 1 and accompanying text at page 6, lines 14-30).

B.   Independent Claim 10

Independent claim 10 relates to a machine-implemented system that stresses a server, comprising an execution engine that generates a scenario that loads the server *via* a plurality of users, the plurality of users dynamically adjusted based on predetermined weightings of a user

profile having weighted characteristics that comprises at least a browser type therein, wherein the scenario distributes user characteristics as a percentage of total requests. Thus, different scenarios and environments can be dynamically created by adjusting the number of users and characteristics to stimulate different stress situations for the server. (*See* Fig. 6 and accompanying text at page 11, lines 9-23).

C.   Independent Claim 16

Independent claim 16 relates to a computer-implemented method for load testing a server comprising assigning weights to user characteristics in a user profile, dynamically adjusting the user characteristics based on one or more browser types during the testing of the server, and distributing the user characteristics as a percentage of total requests sent to the server. This allows an administrator to place controllable amounts of stress on servers for load testing the server. (*See* Fig. 2 and accompanying text at page 7, line 1 – page 8, line 12).

D.   Independent Claim 21

Independent claim 21 relates to a machine-implemented system for test loading a server comprising means for dynamically adjusting user characteristics while loading the server and means for distributing the user characteristics as a percentage of total requests sent to the server, each user characteristic including at least a browser type. In this regard, the server can be test loaded with real user characteristics to simulate an actual usage environment. (*See* Fig. 1 and accompanying text at page 6, lines 14-30).

The means for limitations described above are identified as limitations subject to the provisions of 35 U.S.C. §112 ¶6. The structures corresponding to these limitations are identified with reference to the specification and drawings in the above-noted parentheticals.

VI.   Ground of Rejection to be Reviewed (37 C.F.R. §41.37(c)(1)(vi))

A.   Claims 1-9 and 16-20 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Chen, *et al.* (US 5,812,780) in view of Bernardin, *et al.* (US 2003/0191795).

B.   Claims 10-15 and 21 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Malmskog, *et al.* (US 6,721,686) in view of Bernardin *et al.*

VII.   Argument (37 C.F.R. §41.37(c)(1)(vii))

A.   Rejection of Claims 1-9 and 16-20 Under 35 U.S.C. §103(a)

Claims 1-9 and 16-20 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Chen, *et al.* (US 5,812,780) in view of Bernardin, *et al.* (US 2003/0191795). It is requested that this rejection be reversed for at least the following reasons. Chen, *et al.* and Bernardin, *et al.*, when taken alone or in combination, fail to teach or suggest all elements recited in the subject claims.

> [T]he prior art reference (or references when combined) must teach or suggest all claim limitations. *See* MPEP §706.02(j). *See In re Vaeck,* 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

The subject matter claimed herein relates to loading a test server with simulated user interaction to determine one or more metrics regarding performance of the server. According to one example, different user profiles or characteristics can be utilized in the same tests to better simulate actual user use. To this end, claim 1 recites *dynamically adjusting at least one characteristic of at least one of the simulated users based at least in part on a browser type related to the simulated user, for distribution thereof as a percentage of total requests sent to a server being load tested.* Independent claim 16 recites similar aspects, namely *dynamically adjusting the user characteristics based on one or more browser types during the testing of the server.* Chen, *et al.* and Bernardin, *et al.*, when taken alone or in combination, fail to teach or suggest this aspect.

Chen *et al.* relates to a system for the provision of realistic load conditions on a server application by simulating the behavior of multiple users operating client software. As the Examiner clearly acknowledges, however, Chen *et al.* does not disclose *dynamically adjusting at least one characteristic of at least one of the simulated users based at least in part on a browser type related to the simulated user,* or any behavior regarding a browser type as recited in the subject claims. (*See* Final Office Action dated February 25, 2008). The Examiner offers Bernardin, *et al.* to cure this deficiency; however, Bernardin, *et al.* is similarly lacking.

Bernardin, *et al.* relates to a system for scheduling tasks in a parallel architecture wherein an application program interface can be provided to facilitate remote operation with the

architecture. The task scheduling can be automated based on priority of jobs, for example, and task results can be sent to the requestor of the task. Additionally, Bernardin, *et al.* describes an API for the system that allows for creating an interface that can be utilized by a web browser to manually adjust task scheduling. However, Bernardin, *et al.* fails to disclose or suggest determining characteristics or any characteristics comprising a ***browser type*** of a simulated user, much less dynamically adjusting such characteristics as recited in the subject claims.

In fact, Bernardin, *et al.* is completely silent regarding such aspects. In the sections cited by the Examiner in support, Bernardin, *et al.* discloses functionality of a Broker, which receives task requests and handles scheduling thereof; namely, the section discloses determining "at least one, two, three, four, or more of the following attributes of the available processing resource: (i) operating system of the processing resource; (ii) available memory of the processing resource; (iii) available disk space of the processing resource; (iv) security features of the processing resource; (v) speed of the processing resource; (vi) availability of locally-cached data at the processing resource; (vii) typical frequency of local user activity at the processing resource; and (viii) time of most recent local user activity at the processing resource." (*See* paragraph [0075]).

Additionally, other sections cited by the Examiner discuss resolving scheduling problems, caching, data set utilization, interactive modeling/data visualization, and an administrative interface accessible by compatible browser. (*See* paragraphs [0287], [0326], [0352], and [0361]). Nowhere does Bernardin, *et al.* contemplate adjusting simulated user characteristics ***based on one or more browser types*** or characteristics comprising a ***browser type*** of a simulated user as recited in the subject claims. To the extent the Examiner believes allowing accessibility to an API from different compatible browsers as disclosed by Bernardin, *et al.* recites this aspect, applicants' representative respectfully disagrees. In fact, this appears to be the Examiner's argument according to the Advisory Action dated May 23, 2008. However, allowing access to compatible browsers is not indicative of performing any action (*e.g.*, dynamically adjusting characteristics) or differentiating based on a type of browser as recited in the subject claims, much less even making a determination of browser type to being with. For example, before an action can be performed based on a ***browser type*** as recited in the claims, the browser type must be acquired; both references are silent even in regard to this aspect. Thus, Bernardin, *et al.* does not differentiate between browser types and cannot be said to teach aspects regarding determination of a browser type as recited in the subject claims.

Moreover, the Examiner asserts that combining Chen, *et al.* and Bernardin, *et al.* would "allow an administrator to monitor and manage the server with increasing secured network, and increased convenience to authorized users from any compatible browser." (*See* Final Office Action dated February 25, 2008). Assuming *arguendo* that the combination teaches such aspects, this is not what the applicants are claiming. The claims recite dynamically adjusting simulated user characteristics based on a ***browser type*** of a simulated user, not an administrator monitoring and managing a server *via* a compatible browser. In applicants' claims, the browser type is causing the dynamic adjustment of user characteristics; in Bernardin, *et al.*, however (and the combination with Chen, *et al.*), the browser type is not even considered, as described more fully above. In addition, as Bernardin, *et al.* teaches an API for managing a system, combining Chen, *et al.* and Bernardin, *et al.* would merely produce a web-accessible API for a load server testing application. This combination does not teach or suggest adjusting user characteristics, or anything, based on a browser type as recited in the subject claims.

Furthermore, claim 5 recites *the characteristic **statistically determined based on web log records***. Though Chen, *et al.* discloses utilizing a log file, the log file is used to report statistics of the load testing services. Conversely, claim 5 recites characteristics statistically determined from the log, meaning that not only is the log used post-creation in applicants' claims, but it is utilized to determine characteristics based on the contents. For example, simulated users can be based in part on users discerned from the log file. Chen, *et al.* discloses creating the log file, but is silent in regards to the additional aspects recited in the claims. Thus, the references fail to teach or suggest this additional aspect. The Examiner reiterates the citation to Chen, *et al.* in the Advisory Action, but provides no response or additional argument to the above. Thus, applicants' representative retains the argument that the references fail to teach or suggest all elements of claim 5 as well.

In view of at least the foregoing, it is readily apparent that Chen, *et al.* and Bernardin, *et al.*, when taken alone or in combination, fail to teach or suggest all aspects of claims 1, 5, and 16. Therefore, rejection of these claims, as well as, claims 2-4, 6-9, and 17-20, which respectively depend therefrom, should be reversed.

B.   Rejection of Claims 10-15 and 21 Under 35 U.S.C. §103(a)

Claims 10-15 and 21 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Malmskog, *et al.* (US 6,721,686) in view of Bernardin *et al.* It is requested that this rejection be reversed for at least the following reasons. Malmskog, *et al.* and Bernardin, *et al.*, when taken alone or in combination, fail to teach or suggest all elements recited in the subject claims.

As described, the subject matter claimed herein relates to loading a test server with simulated user interaction to determine one or more metrics regarding performance of the server. According to one example, different user profiles or characteristics, such as browser type, can be utilized in the same tests to better simulate real world user use. To this end, claim 10 recites a *plurality of simulated users dynamically adjusted based on predetermined weightings of a user profile related to at least one of the simulated users having **weighted characteristics that comprises at least a browser type** therein.* Claim 21 recites similar aspects, namely *means for dynamically adjusting characteristics of a simulated user while loading the server.* Malmskog, *et al.* and Bernardin, *et al.*, when taken alone or in combination, fail to teach or suggest this aspect.

Malmskog, *et al.* relates to a system for generating a load on a web server and testing the performance thereof. Specifically, a number of synthetic clients can be produced utilizing various communication metrics, bandwidth, packet loss rate, delay, *etc.* As the Examiner clearly acknowledges, however, Malmskog, *et al.* does not disclose *weighted characteristics comprising at least a browser type* as recited in the subject claims. (*See* Final Office Action dated February 25, 2008). The Examiner offers Bernardin, *et al.* to cure this deficiency; however, Bernardin, *et al.* is similarly lacking as described previously.

In particular, Bernardin, *et al.* does not contemplate simulated user characteristics comprising a *browser type* or utilizing such as recited in the subject claims. As far as allowing accessibility from a compatible browser, as recited in the sections cited by the Examiner, this is not indicative a browser type characteristic of a simulated user that can be used to adjust the simulated user as claimed. Though Bernardin, *et al.* may be operable with multiple compatible browsers, this does not indicate that a browser type is utilized as a characteristic in any way (*e.g.*, that the browser type is determined and used as a simulated user characteristic, much less a weighted one) as in the subject claims. The Examiner contends, in the Advisory Action, that it would have been obvious to modify Malmskog, *et al.* with Bernardin, *et al.* to include

adjustment based on browser type. Applicants' representative respectfully disagrees. First, as described in detail above, Bernardin, *et al.* does not teach or suggest evaluating, taking action, much less adjusting characteristics, based on a browser type of a user. Secondly, assuming *arguendo* that Bernardin, *et al.* did teach these aspects, there is no evident motivation to include such in a system for creating a simulated user environment, as in applicants' claims.

Furthermore, combining Malmskog, *et al.* and Bernardin, *et al.*, if the references are even combinable, produces a web accessible API for the test load system of Malmskog, *et al.* However, the browser type utilized to access the API has no relevance to the system other than compatibility from the accessing user's standpoint.

In view of at least the foregoing, it is readily apparent that Malmskog, *et al.* and Bernardin, *et al.*, when taken alone or in combination, fail to teach or suggest all aspects of claims 10 and 21. Therefore, rejection of these claims, as well as, claims 11-15, which depend therefrom, should be reversed.

C.   **Conclusion**

For at least the above reasons, the claims currently under consideration are believed to be patentable over the cited references. Accordingly, it is respectfully requested that the rejection of claims 1-21 be reversed.

If any additional fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP637US].

Respectfully submitted,
AMIN, TUROCY & CALVIN, LLP

 /David Matthew Noonan/
David Matthew Noonan
Reg. No. 59,451

AMIN, TUROCY & CALVIN, LLP
57th Floor, Key Tower
127 Public Square
Cleveland, Ohio 44114
Telephone: (216) 696-8730
Facsimile: (216) 696-8731

VIII.   Claims Appendix (37 C.F.R. §41.37(c)(1)(viii))

1.   A computer-implemented system that test loads a server comprising:

a dynamic load adjustor component that dynamically adjusts user characteristics based at least in part on a browser type, for distribution thereof as a percentage of total requests sent to a server being load tested.

2.   The system of claim 1, further comprising a profile characteristic data store that supplies the dynamic load adjustor component with weighting for a characteristic defined in a user profile.

3.   The system of claim 2, the dynamic load adjustor component further comprises a weighting designator that randomly assigns to users characteristics based on weightings defined in the user profile.

4.   The system of claim 2, the characteristic comprises at least one of: network connections, browser types, and load patterns.

5.   The system of claim 2, the characteristic statistically determined based on web log records.

6.   The system of claim 2, the characteristic predetermined in a single user profile.

7.   The system of claim 1, further comprising a load coordinator component that adjusts an intensity of a load test based on a current distribution of users entering and leaving the server relative to a desired test load.

8.   The system of claim 1, further comprising an artificial intelligence component.

9.   The system of claim 1, further comprising a closed loop control to enable a continual and sustained rate of requests to the server.

10.   A machine-implemented system that stresses a server, comprising:

an execution engine that generates a scenario that loads the server *via* a plurality of users, the plurality of users dynamically adjusted based on predetermined weightings of a user profile having weighted characteristics that comprises at least a browser type therein, wherein the scenario distributes user characteristics as a percentage of total requests.

11.   The system of claim 10, the scenario comprises at least one of a test mix and a load profile.

12.   The system of claim 10, further comprising a control input that adjusts rate of requests loaded onto the server.

13.   The system of claim 10, further comprising a queuing mechanism that retrieves and sorts requests to be sent to the server.

14.   The system of claim 10, further comprising a scheduler that determines number of requests to be generated for an upcoming period.

15.   The system of claim 10, the requests sorted according to a time function for execution.

16.   A computer-implemented method for load testing a server comprising:

assigning weights to user characteristics in a user profile;

dynamically adjusting the user characteristics based on one or more browser types during the testing of the server; and

distributing the user characteristics as a percentage of total requests sent to the server.

17.   The method of claim 16, further comprising comparing a current load on the server with a desired load.

18.   The method of claim 17, further comprising creating a new user if the current load falls below a desired load.

19.   (Previously Presented) The method of claim 17, further comprising reducing the current load by one upon ending an iteration, if the current load rises above the desired load.

20.   (Previously Presented) The method of claim 16, further comprising controlling a rate of loading *via* a feedback loop control.

21.   A machine-implemented system for test loading a server comprising:
         means for dynamically adjusting user characteristics while loading the server; and
         means for distributing the user characteristics as a percentage of total requests sent to the server, each user characteristic including at least a browser type.

IX.   Evidence Appendix (37 C.F.R. §41.37(c)(1)(ix))
         None.

X.   Related Proceedings Appendix (37 C.F.R. §41.37(c)(1)(x))
         None.